Carnegie-Mellon University
Software Engineering Institute

# Functional Performance Specification
# for an Inertial Navigation System

## B. Craig Meyers

### October 1988

# Functional Performance Specification for an Inertial Navigation System

## B. Craig Meyers

Naval Surface Warfare Center Affiliate

## Nelson H. Weiderman

Real-Time Embedded Systems Testbed Project

This technical report was prepared for the

SEI Joint Program Office
ESD/XRS
Hanscom AFB, MA 01731

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

**Review and Approval**

This report has been reviewed and is approved for publication.

FOR THE COMMANDER

Karl H. Shingler
SEI Joint Program Office

# Table of Contents

# List of Figures

# List of Tables

# Acknowledgement

# Functional Performance Specification
# for an Inertial Navigation System

**Abstract:** This document defines the functional and performance requirements for the inertial navigation system simulator that interfaces with the external computer system (ECS) simulator. Both the INS simulator and the ECS simulator are being developed in Ada by the Real-Time Embedded Systems Testbed Project at the Software Engineering Institute (SEI). The INS simulator is similar to a real-world INS, but has reduced functionality. This document provides specifications for the major functions of the INS simulator.

# 1. Introduction

The purpose of this document is to define a set of requirements applicable to a shipboard inertial navigation system (INS) simulator. An inertial navigation system transmits digital data to external computer systems in a time-critical manner. The data transmitted include ship position, speed, heading, and attitude information. These data are then used by an external computer system for display, aligning missiles, and for planning activities, for example.

The INS that this simulation addresses is the AN/WSN-5, which is an operational system, and which is described in references [3] - [5]. The simulation described in this document is intended to be a realistic subset of the operational AN/WSN-5. That is, the simulation described here must meet the same timing requirements with regard to data updating. Additionally, the simulation must meet the same time-critical demands as the operational AN/WSN-5 regarding transmission and receipt of messages from an external computer. The external behavior of the simulator is identical to the AN/WSN-5.

The INS simulator is being developed by the Real-Time Embedded Systems Testbed at the Software Engineering Institute (SEI). From the perspective of the SEI the goals of this project are to:

1. Develop a real-time program in Ada that may be related to an existing operational system.
2. Have a baseline application Ada program that may be used to test Ada run-time environments.

A system specification document is described in reference [2]. The software for the external computer, which is also developed as part of this project, is described in reference [7].

This document is organized in the following manner: In the next chapter, we present a functional overview of the INS simulator. The detailed functional requirements are presented in Chapter 3, followed by the references. Several appendices are included with this document. The first appendix contains a list of acronyms and symbols referenced in the text. Appen-

dices 2-10 describe the mathematical details involved in the simulation. Appendix 11 contains a discussion of lever-arm constants, which are ship-dependent data used to calculate ship-translational velocities. Finally, Appendix 12 specifies default values of basic parameters as a function of sea state.

# 2. Functional Overview

In this chapter, we give a brief functional overview of the INS simulator, including the purpose and an overview of the required software functionality. We also note the hardware configuration for the initial version of the software.

## 2.1. Purpose

The purpose of a shipboard inertial navigation system is to provide external computer systems with the following ship information:

1. Geographic position (latitude and longitude)
2. Velocity (north, east, vertical)
3. Velocity integrals (north and east)
4. Attitude (roll, pitch, heading)
5. Attitude rate (roll, pitch, heading)

The above information is calculated in a time-critical manner. It is then transmitted via digital messages to external computer systems. These systems use this information for display purposes, planning operations, and performing missile alignment calculations, for example.

## 2.2. Software Functional Overview

The software for the INS simulator may be grouped into the following categories:

1. Initialization functions
2. Operator interface
3. Calculation of simulated ship motion quantities
4. Message processing
5. Built-in test processing
6. Data extraction

Each category is described below.

### 2.2.1. Initialization Functions

There are two functions that are involved in the initialization of the software. A system initialization function is invoked after program initiation and is responsible for all data initializations. A second initialization function is provided by the parameter initialization function, which provides initial values of all required variables that are used in the calculation of simulated quantities.

## 2.2.2. Operator Interface

The operator interface function provides the capability for display of values to the operator and for operator input of quantities to the simulation program. The processing for the operator interface is divided between a display function and an input function. The basic screen display is shown in Figure 1.

```
┌─────────────────────────────────────────────────────────────┐
│                                                             │
│                                                             │
│                                                             │
│                                                             │
│                    Status  Display  Area                    │
│                                                             │
│                                                             │
│                                                             │
│                                                             │
├─────────────────────────────────────────────────────────────┤
│                   Input  and  Display  Area                 │
├─────────────────────────────────────────────────────────────┤
│                      Alert  Display  Area                   │
├─────────────────────────────────────────────────────────────┤
│  DX:                                        EC Status:      │
└─────────────────────────────────────────────────────────────┘
```
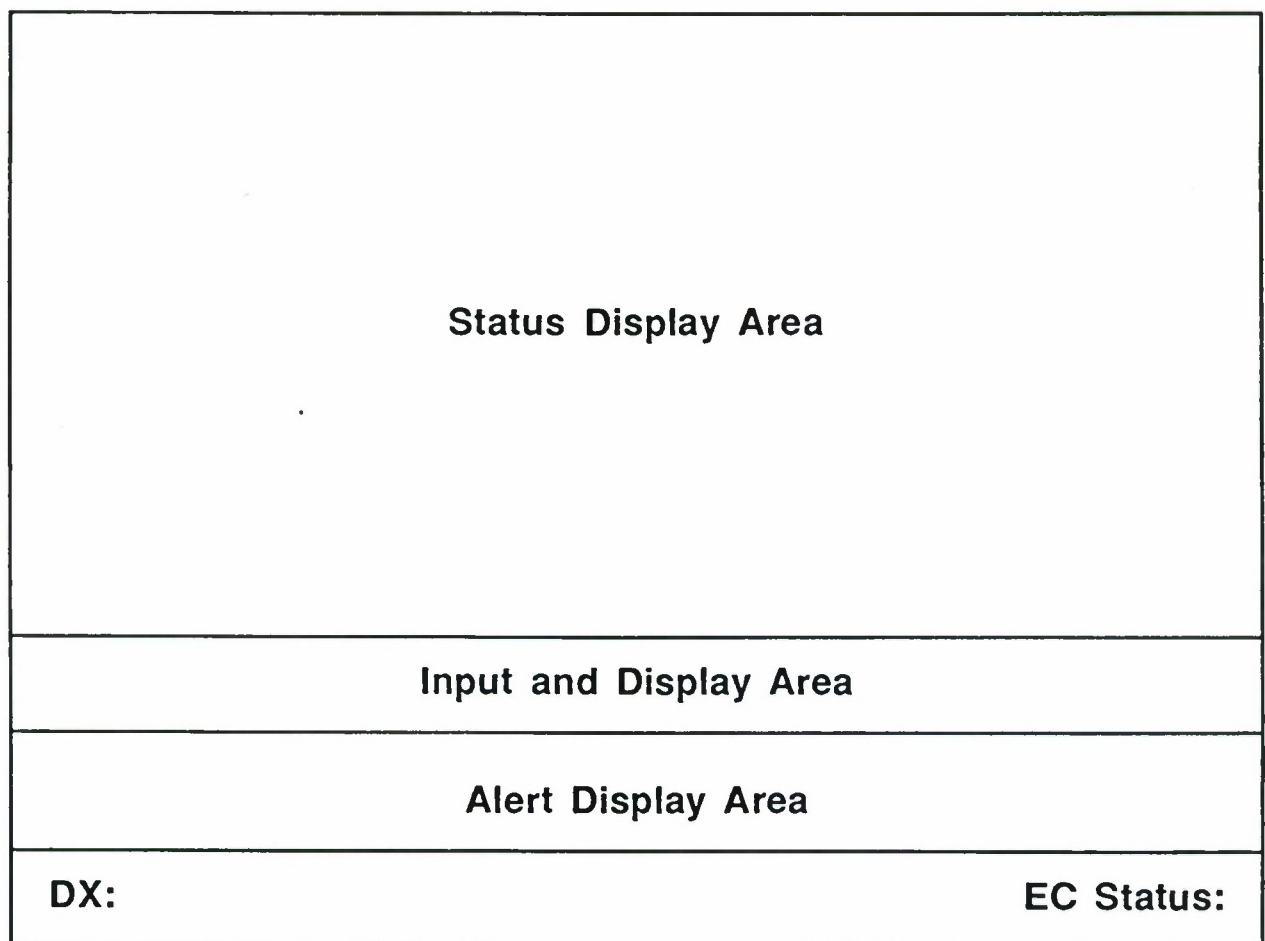
**Figure 1:** INS Operator Console Screen Allocation

The upper part of the screen is a status display area that is reserved for the display of calculated quantities and is updated at a rate of once per second. Below this is the input display area. This part of the screen is reserved for the operator entry of commands and display of certain quantities. An alert display area is also present on the screen and is used for the display of alerts to the operator. Finally, the last part of the screen is reserved for the display of information about data extraction and the state of the INS-external computer interface.

The input function of the operator interface function processes all input from the operator. The commands that may be entered can be grouped into the following categories:

1. **Parameter Definitions.** The operator has the ability to change the basic parameters which are used in the calculation of simulated quantities. Thus, for example, the operator may change the value of the amplitude of the ship roll.

2. **Fault Injections.** The operator may also change the values of the quantities that are output by the INS simulator. This fault injection capability is included to test the response of the external computer to values that are out of the specified range.

3. **Commands.** The operator may also specify commands that are used to determine the processing related to certain simulated quantities. Thus, for example, the operator may issue a command to change the ship course or to increase the speed of the ship over a certain time interval. This information is then used in the calculation of the simulated quantities that are output by the INS.

4. **Data Extraction.** The operator may specify that the data extraction capability be enabled or disabled. Additionally, the name of the file where the extracted data will reside may also be specified.

The ability of an operator to issue commands as illustrated above, as well as to provide a mechanism for fault injection, increases the versatility of the INS simulator. Thus, these latter types of input are particularly useful when the INS simulator is to be used to "drive" an external computer system.

### 2.2.3. Calculation of Simulated Ship Motion Quantities

The motion of a ship is described in terms of a six-degrees-of-freedom model. The translational motion of the ship is described by surge, sway, and heave motion. The rotational motion of the ship is described by roll, pitch, and yaw motions. These six quantities are calculated based on the following assumptions:

1. The motion of the ship in any degree of freedom may be described by a sinusoidal motion.

2. The motion of the ship in any degree of freedom is uncoupled from the motion in any other degree of freedom.

More elaborate models of ship motion are possible, but one based on the above assumptions is adequate. That is, the principal purpose of this work is to: (i) explore the application

of Ada to real-time systems; and (ii) develop an INS simulator that may be used in connection with the development and testing of other systems. Therefore, the principal functional emphasis of this document is to faithfully implement the interface processing of digital messages to an external computer system.

### 2.2.4. Message Processing

A major element of the INS simulator is to provide an external computer system with time-critical values related to ship motion. These values are transmitted by digital messages. The message processing function supports all processing for intercomputer communication. This includes functions for enabling communications, input message processing, output message processing, and disabling communications.

### 2.2.5. Built-In Test Processing

The software includes two functions related to built-in test (BIT) processing. The first of these is executed upon program initiation and performs various checks of the machine hardware to assure the integrity of the system. The initial BIT test also includes a program checksum to assure a successful program load. A second type of run-time BIT processing is performed as a background task during real-time execution. This BIT performs reasonableness checks on the contents of those memory addresses used to store output message buffers.

### 2.2.6. Data Extraction

A limited data extraction capability is provided in the software. This allows the system to extract, during real-time execution, the values of certain parameters. These values may then be used during the testing of the software. There is no requirement that the data extraction capability record any message transmissions. Such information is available for testing from an external measuring system, as noted in Section 2.3.

### 2.2.7. Language Requirement

With exceptions noted below, all software to be developed for the INS simulation shall be in Ada, in accordance with reference [1]. The exceptions to this are that software for the built-in test functions as well as low-level I/O drivers may be written in assembler. Additionally, the use of Ada may be waived for bit manipulation and conversion routines that relate to compiler-specific processing, as may be necessary, for example, In relation to message encoding and decoding.

## 2.3. Hardware Configuration

The hardware configuration for the systems to be developed is shown in Figure 2. All systems shall be resident in MicroVAX-2 processors with VT-220 terminals used as the operator interface. Communication between the INS and the external computer shall be via 16-bit parallel interfaces.

Also shown in Figure 2 is an external monitoring system that interfaces between the INS and the external computer. The purpose of the external monitoring system is to be able to "capture" data in real time as they are transmitted across the interface. As data are collected by this system, they shall be time-tagged to an accuracy of at least one millisecond. The data collected by the external monitoring system will be used in the system testing phase of the project.

```
┌─────────────┐                    ┌─────────────┐
│     INS     │════════════════════│  External   │
│  Simulator  │         │          │  Computer   │
│             │         │          │  Simulator  │
└──────┬──────┘         │          └──────┬──────┘
       │                │                 │
    ┌──┴──┐             │              ┌──┴──┐
    │ ╭─╮ │             │              │ ╭─╮ │
    │ ╰─╯ │             │              │ ╰─╯ │
    └─────┘             │              └─────┘
  Operator             │            Operator
  Console          ┌───┴───┐        Console
                   │External│
                   │Monitor.│
                   │ System │
                   └────────┘
```
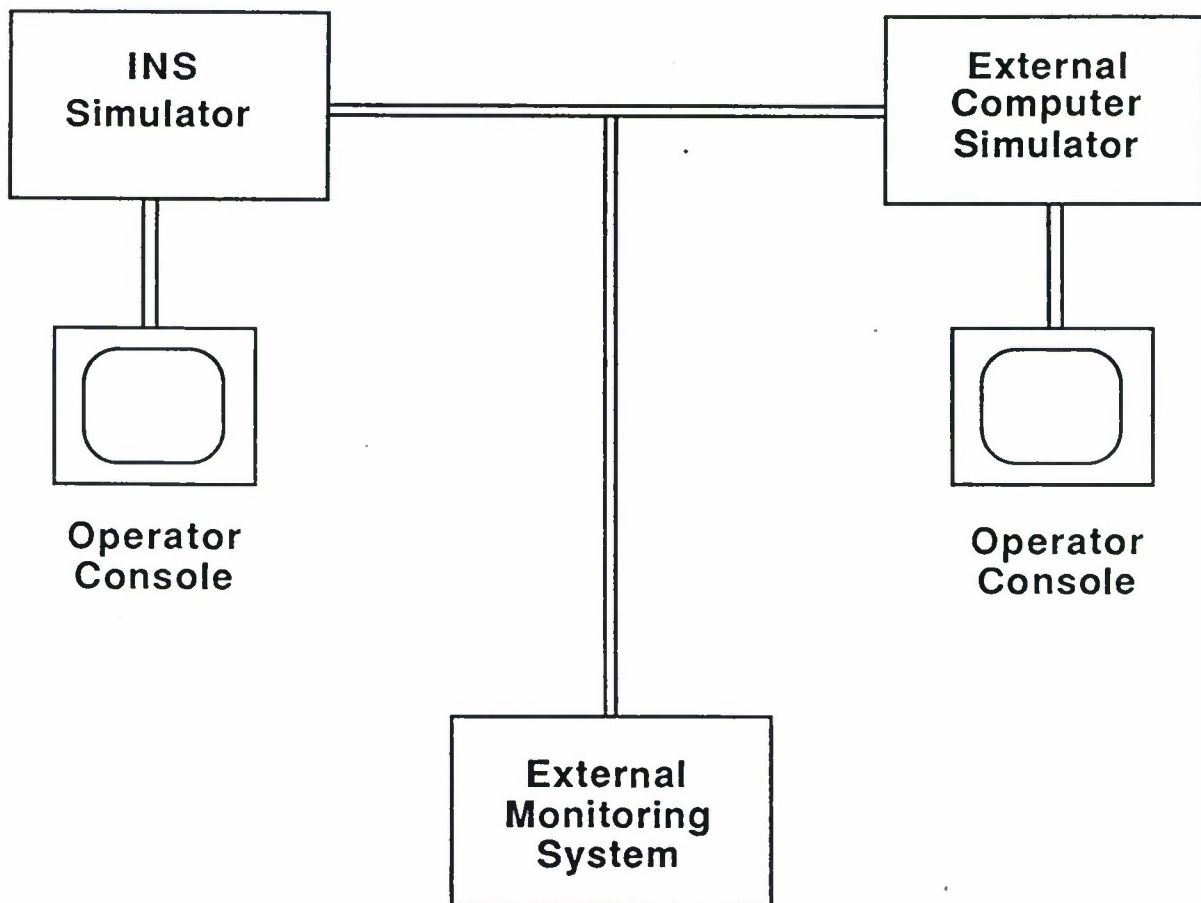
**Figure 2:** Hardware Configuration

# 3. Detailed Functional Requirements

The detailed functional requirements applicable to the INS simulator are described in this chapter.

## 3.1. System Initialization Function

The purpose of the system initialization function is to perform all initializations required for beginning the simulation. When initiated, the following processing shall take place:

1. All data values shall be initialized to their proper values.
2. The time of gyro reset shall be set to the current clock time.
3. The status of communications between the INS and the external computer shall be set to down.

## 3.2. Parameter Initialization Function

The purpose of the parameter initialization function is to allow the operator to define all parameters necessary for determining the quantities that are simulated by this program. These include initial values of parameters, values of parameters used to calculate ship roll, pitch and yaw, lever arm constants, and other values. The parameters are a function of sea state.

This function shall be initiated upon completion of a successful initial BIT. When initiated, this function shall provide for initial default values for the quantities specified below.

1. Latitude and longitude
2. Commanded course
3. Speed
4. Ship list and trim
5. Quantities necessary to calculate ship roll; see Appendix 2.
6. Quantities necessary to calculate ship pitch; see Appendix 3.
7. Quantities necessary to calculate ship yaw; see Appendix 4.
8. Quantities necessary to calculate ship surge; see Appendix 5.
9. Quantities necessary to calculate ship sway; see Appendix 6.
10. Quantities necessary to calculate ship heave; see Appendix 7.
11. Lever arm constants; see Appendix 11.

The default values of the above parameters for the different sea state conditions are specified in Appendix 12. The values shall be initialized as a function of sea state and shall provide for sea state values in the range from one to seven.

As each parameter is accessed, it shall be verified to be within a legal range. The legal ranges of the above parameters are described in Section 3.3.2.2. If a parameter is found to be out of bounds, an alert will be issued to the operator and processing by this function shall terminate. When all parameters have been defined, an alert message shall be displayed to the operator to this effect. Following this, the parameter initialization function shall be terminated.

Once a parameter file has been loaded, the values of parameters may be changed by the operator; see Section 3.3.2.2.

## 3.3. Operator Interface Function

The interaction between the operator and the INS simulator is through the operator interface function. This function is described in terms of an operator display function and an operator input function. Each of these is described below. A related function, the alert processing function, is discussed in Section 3.4.

### 3.3.1. Operator Display Function

A part of the screen display is reserved for the display of information concerning the values of system variables. This area, denoted in Figure 1 as the status display area, occupies the upper portion of the screen.

The operator display function shall update the status display area of the screen at a rate of once per second. The format of the simulated quantities to be displayed is indicated in Figure 3. In this figure, GMT and TGR denote Greenwich Mean Time and Time of Gyro Reset, respectively.

```
┌──────────────────────────────────────────────────────────────────┐
│  Lat:      xx xx.xx                    GMT: xx xx xx               │
│  Long:  xxx xx.xx                      TGR: xx xx xx               │
│                                                                    │
│                                        Speed:   xx  knots          │
│                                                                    │
│  Course:      xxx.xx   Deg                                         │
│ ·Heading:     xxx.xx   Deg                                         │
│  Roll:         xx.xx   Deg    Rate:      xxx.xx Deg/Sec            │
│  Pitch:        xx.xx   Deg    Rate:      xxx.xx Deg/Sec            │
│  Yaw:          xx.xx   Deg    Rate:      xxx.xx Deg/Sec            │
│                                                                    │
│  Surge:   xx.xx   Ft     Sway: xx.xx  Ft    Heave: xx.xx Ft       │
│  List:        xx.xx   Deg              Trim: xx.xx Deg            │
│  Ocean (East):  xx.xx   knots   Ocean (North): xx.xx knots        │
│                                                                    │
│  Vel East:    xx.xx knots    Cumulative:    xxxxxxx  Ft           │
│  Vel North:   xx.xx knots    Cumulative:    xxxxxxx  Ft           │
│  Vel Vert:    xx.xx knots                                         │
│                                                                    │
├──────────────────────────────────────────────────────────────────┤
│                                                                    │
├──────────────────────────────────────────────────────────────────┤
│  EC Communications UP                        xx xx xx             │
├──────────────────────────────────────────────────────────────────┤
│  DX:  OFF                              EC Status:  UP             │
└──────────────────────────────────────────────────────────────────┘
```

**Figure 3:**   Sample Operator Console Layout

## 3.3.2. Operator Input Function

This function is responsible for the input of data and commands from the operator. The discussion of the requirements of this function are organized in terms of applicable commands, and are listed below.

### 3.3.2.1. General Requirements

The following general requirements apply to the input of data by the operator:

1. Input shall not be case sensitive. That is, either upper or lower case characters shall be accepted by this function.
2. White space, such as blanks and tabs, shall be ignored when present in the input data.

### 3.3.2.2. Setting Parameter Values

The operator shall have the ability to change values of parameters that are used in the calculation of simulated quantities. The form of this command shall be:


SET <PARAMETER_NAME> = <PARAMETER_VALUE>


In the above, <PARAMETER_NAME> represents the name of the parameter that is to be changed. The <PARAMETER_VALUE> indicates the new value that is to be assigned to the parameter.

When the above command is recognized, the <PARAMETER_NAME> shall be verified. Also, the requested new <PARAMETER_VALUE> shall be verified to be within the range of legal values. Table 1 provides a list of the valid parameter names and the range of legal values. If an error is detected, an alert shall be issued to the operator and the parameter change request shall be ignored.

The actual value of a <PARAMETER_NAME> is not changed until the operator has entered the character string 'ENTER'. This permits the simultaneous changing of several parameters. Upon completion of the changes, an alert shall be issued to the operator.

### 3.3.2.3. Display of Parameter Values

The operator shall be able to obtain the current value for a parameter by a command of the form:


SHOW <PARAMETER_NAME>


where <PARAMETER_NAME> is a a valid parameter name; see Table 1. When this command is recognized, the value of the requested parameter shall be displayed in the input display area of the screen. Errors detected in processing this command shall be processed as described in Section 3.3.2.2.

The software shall also provide the capability to display all current parameters and associated values. If this request is issued by the operator, the parameters shall be displayed in the status display area indicated in Figure 1. When such a request has been issued, the software shall suspend the operator display function specified in Section 3.3.1. Additionally, the software shall provide a capability to restore the periodic display processing.

---

**Table 1:** Parameter Names and Legal Ranges
That May Be Changed
by the Operator Using the "Set" Command

| | | | |
|---|---|---|---|
| HEAVE_AMP | $A_{heave}$ | FT | [0, 11] |
| HEAVE_FREQ | $\omega_{heave}$ | RADIANS | [0, 0.4] |
| HEAVE_PHASE | $\phi_{heave}$ | DEG | [0, 360] |
| LAC_A | a | FT | [-250, 250] |
| LAC_B | b | FT | [-25, 25] |
| LAC_C | c | FT | [-25, 25] |
| LIST | $\theta_{list}$ | DEG | [-2, 2] |
| PITCH_AMP | $A_{pitch}$ | DEG | [0, 3] |
| PITCH_FREQ | $\omega_{pitch}$ | RADIANS | [0, 0.4] |
| PITCH_PHASE | $\phi_{pitch}$ | DEG | [0, 360] |
| ROLL_AMP | $A_{roll}$ | DEG | [0, 28.6] |
| ROLL_FREQ | $\omega_{roll}$ | RADIANS | [0, 0.4] |
| ROLL_PHASE | $\phi_{roll}$ | DEG | [0, 360] |
| SHIP_COURSE | $\beta$ | DEG | [0, 360] |
| SURGE_AMP | $A_{surge}$ | FT | [0, 10] |
| SURGE_FREQ | $\omega_{surge}$ | RADIANS | [0, 0.4] |
| SURGE_PHASE | $\phi_{phase}$ | DEG | [0, 360] |
| SWAY_AMP | $A_{sway}$ | FT | [0, 15] |
| SWAY_FREQ | $\omega_{sway}$ | RADIANS | [0, 0.4] |
| SWAY_PHASE | $\phi_{sway}$ | DEG | [0, 360] |
| TRIM | $\theta_{trim}$ | DEG | [-2, 2] |
| YAW_AMP | $A_{yaw}$ | DEG | [0, 5] |
| YAW_FREQ | $\omega_{yaw}$ | RADIANS | [0, 0.4] |
| YAW_PHASE | $\phi_{yaw}$ | DEG | [0, 360] |

---

### 3.3.2.4. Fault Injection

The operator shall have the ability to inject faults. That is, the operator shall be able to change the values of certain quantities that are output by the INS. The format of this command shall be:

FAULT <VARIABLE_NAME> = <FAULT_VALUE>

---

The name of the variable that is to be changed is specified by <VARIABLE_NAME>. The operator shall have the ability to change any message-dependent quantity—that is, any data field that is transmitted in a message to the external computer.

A check shall be performed to assure that the requested fault value is within a specified range. The variables that may have injected faults are all defined in terms of message-dependent quantities. Hence, the permitted range of a fault variable is that the requested value must be able to be allocated to the storage area in the output message. The permitted range may be determined by reference to the interface design specification, reference [5].

The operator is also required to use the 'ENTER' command to affect the actual fault changes. The actual changes shall not be made until the operator has entered the command 'ENTER' in a manner similar to that described in Section 3.3.2.2.

Errors found when processing this command shall be treated in a manner discussed in Section 3.2.2.2.

### 3.3.2.5. Commands Related to Ship Course
The operator shall have the ability to simulate ship course changes. This shall be accomplished by a command of the form:


TURN TO {PORT | STARBOARD} AT <TURN_RATE> UNTIL COURSE <NEW_COURSE>

The expression for <TURN_RATE> is understood to be in degrees per second and shall be in the range [0, 2]. The expression for the <NEW_COURSE> is understood to be in degrees and shall be in the range [0, 360]. Any errors detected in processing this command shall cause an alert to be issued to the operator, and the requested change shall be ignored.

### 3.3.2.6. Commands Related to Ship Speed
The operator shall have the ability to simulate the commanded ship speed. This shall be accomplished by a command of the form:


{INCREASE | DECREASE} SPEED TO <SPEED_VALUE> IN <TIME_PERIOD>

The expression for <SPEED_VALUE> is understood to be In knots and shall be in the range [0, 40]. The expression for the <TIME_PERIOD> is understood to be in minutes and shall be in the range [0, 120].

Any errors detected in the processing of this command shall result in an alert being issued to the operator, and the requested change shall be ignored.

### 3.3.2.7. Ability to Reset Gyro

The operator shall have the ability to simulate a reset of the gyro by issuing a command of the form: 'RESET GYRO'. When this command is recognized, the time of gyro reset shall be set to the current system time.

### 3.3.2.8. Commands Related to Data Extraction

The operator shall have the ability to enable and disable the data extraction capability. Additionally, the operator shall be able to specify the name of the file that will be used for recording the extracted data. Any error detected during this processing shall result in an alert being issued to the operator.

## 3.4. Alert Processing Function

The purpose of the alert processing function is to be able to notify the operator of information concerning system status and error indications during execution. This function is initiated by other functions when an alert condition has been raised. All alerts shall be maintained in a prioritized list.

When this function is initiated, it shall add the referenced alert to the prioritized alert list. The time of receipt of the alert shall also be maintained by this function. If there is no currently displayed alert, the referenced alert text and alert time shall be displayed in the alert display area of the operator console (see Figure 2). If there is an alert currently displayed, the number of pending alerts shall be displayed in the alert display area. Additionally, an audible alarm shall be generated for a period of two seconds.

If an alert is currently displayed and other alerts are pending display, the operator can press the ESC key to display the next pending alert in the prioritized list.

A list of the alert text strings appears in Table 2.

At a minimum, the system shall provide for maintaining a list of at least 50 alerts. If the list of pending alerts is full and there is another alert to be added to the list, it shall be inserted in a priority order, thereby replacing the lowest priority alert currently on the alert list.

## 3.5. Database Update Function

The purpose of the database update function is to affect the changes in parameters and/or values requested by the operator. This function is initiated from the operator interface function upon successful receipt of an 'ENTER' command following one or more 'SET' commands.

If the values of one or more variables have been changed, this function shall verify that the values are within the legal range. If a value is found to be outside the legal range, an alert shall be issued to the operator and that parameter will not be changed. After the values

have been verified, the database shall be updated and an alert issued to the operator, indicating that the requested change has been made. Note that this function is not to be initiated regarding fault injections. That is, if an operator injects a fault, the value of the fault shall not be updated by this function.

If the Data Extraction capability has been enabled by the operator, this function shall extract the time, variable name, and value of all variables which have been changed. The extracted time to be recorded for variables that are changed shall be the time of the 'ENTER' command.

The following is a list of all alerts which may be displayed by the alert processing function. The alerts are listed below in decreasing order of priority.

---

**Table 2:** List of Alerts

Runtime Bit Failure
Initial Bit Register Test Failure
Initial Bit Address Read/Write Failure
Initial Bit Arithmetic Test Failure
Initial Bit Memory Checksum Failure
Initial Bit Test Successful
Fault Changes Completed
Invalid Message Type in Message
Invalid Number of Words in Message
Invalid Test Pattern Received
EC Communications Up
EC Communications Down
EC Communications Enabled
Select Message Received from EC
Invalid Course Change
Invalid Speed Change
Invalid Turn Command
Invalid DX File Specified
Unable To Open DX File
DX Disk File Write Error
Parameter Initialization Complete
Parameter Changes Completed
Invalid Set Parameter Change Request
Invalid Show Parameter Request
Invalid Fault Request
Invalid Data Extract Request
Invalid Enter Command
Invalid Entry

---

## 3.6. Ship Speed Update Function

The purpose of the ship speed update function is to periodically update the value of the current ship speed. This function shall be initiated every 40.96 milliseconds.

The change in ship speed is determined by the operator command for changing ship speed (see Section 3.3.2.6). If such a command has been entered, this function shall calculate the current ship speed based on the values entered by the operator.

## 3.7. Ship Attitude Update Function

The purpose of the ship attitude update function is to provide time-critical values of ship attitude data: ship roll, ship pitch, and ship yaw. This function shall also calculate values of the ship roll rate and ship pitch rate. This function shall be initiated every 2.56 milliseconds.

When initiated, the following processing shall be performed:

1. The value of ship roll shall be calculated as described in Appendix 2.
2. The value of ship roll rate shall be calculated as described In Appendix 2.
3. The value of ship pitch shall be calculated as described in Appendix 3.
4. The value of ship pitch rate shall be calculated as described in Appendix 3.
5. The value of ship yaw shall be calculated as described in Appendix 4.

Upon completion of the above, processing by this function shall be terminated.

## 3.8. Ship Displacement Update Function

The purpose of the ship displacement update function is to update the values of the ship surge, sway, and heave on a time-critical basis. This function shall be initiated every 40.96 milliseconds.

Upon initiation, the following processing shall be performed:

1. The ship surge shall be calculated as described in Appendix 5.
2. The ship sway shall be calculated as described in Appendix 6.
3. The ship heave shall be calculated as described in Appendix 7.

Upon completion of the above calculations, processing by this function shall be terminated.

## 3.9. Ship Heading Update Function

The purpose of the ship heading update function is to periodically update the values of the ship heading and heading rate. This function shall be initiated every 2.56 milliseconds.

The equations necessary to calculate the ship heading and heading rate are described in Appendix 8. If the ship is currently executing a turn via an operator request (see Section 3.3.2.5), the equations described in Appendix 8 represent a piecewise linear approximation to the continuous change in heading.

## 3.10. Ship Position Update Function

The purpose of the ship position update function is to periodically update the current value of ship position. This function shall be initiated once every 1.30 seconds.

The ship position is a function of the commanded course, speed, previous position, and the time interval between updates. The details of the calculation are described in Appendix 9.

## 3.11. Ship Velocity Update Function

Ship velocities are measured relative to a (north, east, down) coordinate system. It is the purpose of the ship velocity update function to provide time-critical values of the ship translational velocity. Additionally, this function shall provide cumulative measures of the east and north velocity integrals. This function shall be initiated once every 40.96 milliseconds.

The computation of the ship translational velocity is a function of the following:

1. The commanded horizontal speed and course of the ship.
2. The translational velocity of the ship at its center of gravity due to surge, sway, and heave motions.
3. The translational velocity of the INS due to the collective ship roll, pitch, and yaw motion about the ship's center of gravity. This term must also take into account that the INS is not at the ship's center of gravity.

The equations to determine the ship velocity components are described in Appendix 10.

## 3.12. Message Processing Function

The purpose of the message processing function is to perform all processing relevant to transfer of digital messages between the INS and an external computer. It is by the transfer of such messages that the INS provides timely values of quantities describing the ship motion.

The transfer of data messages is accomplished by the use of protocols. The implementation of the protocol is through the use of external function (EF) messages. Some of the EFs are as follows:

1. Acknowledge (ACK)
2. End of Message (EOM)
3. Initialization (ATTN2)
4. Not Acknowledge (NAK)
5. Ready To Receive (RTR)
6. Start of Message (SOM)
7. Start of Test Message (SOTM)
8. Time Out (ATTN1)

The format and content of all intercomputer messages are discussed in detail in the interface design specification for the INS and external computer; see reference [5]. In addition, this reference defines requirements for protocol procedure and other processing relating to the exchange of messages between the INS and the external computer. The requirements specified in reference [5] are applicable to this document. Note, however, that there is no requirement that the software support processing for satellite update messages, described In reference [5].

In the following subsections an overview of the message processing requirements is presented, and the processing whose details are specified in reference [5] is outlined.

## 3.12.1. Enable Communications Function

The external computer is responsible for enabling communications with the INS. The details of this processing are specified in reference [5] and include the following:

1. The external computer transmits a test message to the INS.
2. The INS transmits a test message to the external computer.
3. The INS transmits a time and status data message to the external computer.

Upon successful completion of the above, the communications between the INS and the external computer are enabled. The time and status data message is used to establish a common time reference between the INS and the external computer.

## 3.12.2. Test Message Processing

To verify the integrity of Intercomputer communications, provision shall be made for the exchange of test messages between the INS and the external computer. This processing is initiated by the external computer and includes the following:

1. The external computer shall transmit a test message to the INS.
2. The INS shall validate that the message is received without error. If an error is detected, error processing shall be invoked, and the external computer shall attempt to enable communications.

3. If the message is received without error, the INS shall transmit a test message to the external computer.

4. The external computer shall verify receipt of the test message from the INS. If the message is received without error, the external computer shall transmit an ACK enable function, and the test message processing is complete.

5. If the external computer detects an error in the received test message from the INS or if a timeout occurs, the external computer shall transmit a NAK enable function and shall attempt to reinitialize communications with the INS.

### 3.12.3. Input Message Processing Function

The purpose of the input message processing function is to perform the processing relative to the receipt of messages from the external computer. The following messages may be transmitted by the external computer:

1. External Functions
2. Test Message
3. Select Data Message

The following sections provide a discussion of the message transfer sequence and the message-specific processing. There is no requirement that this simulator shall be able to process any NAVSAT data message as described in reference [5].

#### 3.12.3.1. Input Message Transfer Sequence

The transfer of a message from the external computer to the INS involves the transfer of protocol messages, as well as the actual data message. In the following section, a discussion of the nominal message transfer sequence is presented. This is followed by a discussion of the processing for erroneous conditions.

#### 3.12.3.2. Nominal Data Transfer Sequence

The nominal transfer of a message from the external computer to the INS involves the following basic steps:

1. The external computer transmits a start of message (SOM) external function, indicating the start of a message sequence.

2. The INS transmits a ready to receive (RTR) external function, indicating to the external computer that the INS is ready to accept data.

3. The INS starts a 10.24 millisecond timer awaiting receipt of a response from the external computer.

4. The external computer transmits the data message.

5. The external computer transmits an end of message (EOM) external function.

6. The INS validates the contents of the received message.

7. The INS transmits an acknowledge (ACK) external function.

The above represents the expected sequence of events involved in the transmission of data from the external computer to the INS. Possibilities that errors may be detected in the process are discussed below.

### 3.12.3.3. Erroneous Conditions

The details of the recognition of erroneous conditions and the requisite processing are specified in reference [5]. An overview of the processing for erroneous conditions is presented in the following:

1. As part of the message transfer sequence, the INS uses a 10.24 millisecond timeout. If the timeout expires or an ATTN1 external function is received by the INS, the message sequence is terminated and an alert is issued to the INS operator.

2. If the message validation fails, the INS assumes a bad data transmission. In this case, the data received from the external computer shall not be used, and the INS transmits a negative acknowledge (NAK) external function to the external computer. An alert shall also be issued to the INS operator.

3. The INS validates the messages received from the external computer. As a minimum, this shall include that:

   - The message type received shall be a valid message type.
   - The number of words in the message shall be correct for the message type received.
   - The test word patterns in the message shall be valid.
   - The transfer of data from the external computer shall comply with data transfer protocols.

### 3.12.3.4. Message-Specific Processing

The message-specific processing that is performed by the input message processing function relates to the receipt of a select data message from the external computer. This message is used to select the message type(s) that are to be transmitted by the INS. In particular, the following requirements are applicable when a select data message is received by the INS:

1. If the type of message(s) requested by the external computer is identical to the message(s) currently being output, the message shall be ignored.

2. If the data identification field in the select data message indicates that the INS is to disable output, communications between the INS and the external computer shall be disabled.

3. The type of message(s) requested by the external computer, as indicated by the data identification field of the message, shall be saved.

When a change in the message type(s) requested by the external computer has been recognized, the INS shall comply in transferring the requested message types.

## 3.12.4. Output Message Processing Function

The purpose of the output message processing function is to perform all processing relative to transmission of messages, as well as protocols, to an external computer. The following messages may be output by the INS:

1. External functions

2. Attitude data periodic message

3. Navigation data periodic message

4. Time and status data message

5. Test message

The following sections describe the message transfer sequences and the message-specific processing performed by this function.

### 3.12.4.1. Output Message Transfer Sequence

The transfer of a message from the INS to the external computer involves a transmission of the actual data message as well as certain protocol information. In the following, the nominal output message sequence is defined, and processing for erroneous conditions is discussed.

### 3.12.4.2. Nominal Message Transmission

The nominal sequence of message transmission from the INS to the external computer involves the following steps:

1. The INS transmits a start of message (SOM) external function (EF). This indicates that the INS is ready to transmit a data message to the external computer.

2. The INS starts a 10.24 millisecond timer awaiting a response from the external computer.

3. The external computer transmits a ready to receive (RTR) EF. This indicates that the external computer is ready to accept data from the INS.

4. The INS transfers the data message to the external computer.

5. The external computer validates the message received from the INS.

6. The INS transmits an end of message (EOM) EF to the external computer. This indicates that the actual data message transmission is complete.

7. The INS starts a 10.24 millisecond timeout awaiting a response from the external computer.

8. The external computer transmits an acknowledge (ACK) EF indicating successful receipt of the message from the INS.

Upon receipt of the ACK EF from the external computer, the transmission of the data message sequence terminates. The above represents the expected message transfer sequence. In the following, recognition of erroneous conditions is noted.

### 3.12.4.3. Erroneous Conditions

It is recognized that erroneous conditions may arise in the data transfer sequence. The details of the erroneous conditions and the processing required are described in reference [5]. A brief overview of the erroneous conditions includes the following:

1. The INS may time out while awaiting a response from the external computer. In this case, the INS shall attempt one retransmission of the message, and the INS operator shall be alerted.

2. The response of the external computer may indicate an error was detected. In this case, the external computer may respond with a negative acknowledge (NAK). If this is received by the INS, the INS shall attempt one retransmission of the message, and the INS operator shall be alerted.

3. As indicated above, it is the responsibility of the external computer to validate messages received from the INS. Typically, this includes the following:

   - All messages received by the external computer shall have a valid message type.
   - All messages received by the external computer shall have a correct number of words for the received message type.
   - The basic test word patterns contained in each message shall be correct.
   - The message transmission shall comply with data transfer and message sequence protocols.

### 3.12.4.4. Message-Specific Processing

The following requirements are also applicable to this function:

1. The output of either the attitude data periodic message and/or the navigation data periodic message is determined by receipt of the select data message from the external computer.

2. If selected, the attitude periodic data message shall be transmitted once every 61.44 milliseconds.

3. If selected, the navigation data periodic message shall be transmitted once every 983.04 milliseconds.

4. In the event that more than one message is due for output to the external computer at the same time, the priority is as follows:

   - attitude data periodic message
   - navigation data periodic message
   - time and status data message

## 3.12.5. Disable Communications Function

The purpose of the disable communications function is to terminate interface communication with the external computer. This function is initiated upon receipt of an ATTN4 enable function from the external computer.

Upon initiation, the following processing shall be performed:

1. The INS shall cease to transmit messages (and protocols) to the external computer.

2. An alert shall be issued to the operator indicating that communications with the external computer are disabled.

3. The state of the INS external computer interface shall be set to disable.

## 3.13. Initial BIT Processing Function

The purpose of the initial BIT processing function is to perform certain checks on the hardware to assure the integrity of the machine. This function is initiated following the program load. At a minimum, the following checks shall be made:

1. **Register checks.** Each register in the machine shall be checked to assure that load, store, and shift operations are performed without error.
2. **Address read/write.** Each address within the program address space shall be subject to a read/write verification test. The initial value at each address shall be restored following completion of the test.
3. **Arithmetic capability.** The arithmetic capability of the processor shall be checked by computing the values of the sine and cosine of an angle and the square root function such that the resulting functions should return a similar value. The computed values shall be compared to the expected result, calculated independently, and may be stored as a constant. This test shall determine that the values are all within a specified tolerance.
4. **Memory checksum.** The contents of each address within the program address space shall be checksummed. The result of the checksum shall be compared to a precomputed value.

A failure of any of the tests above shall result in initiation of the alert processing function. This shall result in the display of the failed test for the operator and may be used in diagnostic procedures. If all of the BIT checks pass, an alert will be issued to the operator to this effect, and processing by this function shall be terminated.

## 3.14. Runtime BIT Processing Function

The software shall support a runtime built-in test (BIT) function whose purpose is to test the values of selected variables on a periodic basis. This function shall be initiated at least once per second while communication with the external computer is enabled.

When initiated, this function shall test values contained in the output message buffers that are used for transmission of the attitude data periodic message and the navigation data periodic message. This function shall verify that the message contents lie within acceptable bounds as defined in the interface description document for the INS external computer, reference [5]. If any value is found to be in error, an alert will be issued to the operator.

If the data extraction capability is enabled, the time this function was initiated shall be recorded.

## 3.15. Data Extraction Function

The purpose of the data extraction function is to be able to record the times and values of certain variables during the course of program execution. The extracted data are recorded on an external media and are later available for testing the execution of the program. The data extraction function shall be able to record certain parameters of the program. It is enabled and disabled on a global basis; that is, there is no selective data extraction capability. There is no requirement that the data extraction function record messages and protocols that are sent/received by the INS. Such data may be available through the use of external monitoring devices, however.

The data extraction function is enabled by the operator input function and invoked by other functions. When enabled, the following requirements are applicable:

1. If the data extract capability is on (off) and the request is to turn the extract capability on (off), an alert shall be set and the request ignored.

2. If a valid request is received to enable the data extraction capability, the name of the data extract file specified by the operator shall be verified. If the file name is invalid, an alert shall be issued to the operator and the request ignored.

3. If a valid request is received to enable the data extraction function and the file name is valid, the specified file shall be opened. If an error is encountered opening the file, an alert will be issued to the operator, the request ignored, and the state of data extraction capability shall be set to an error state.

The data extraction function shall record the following information:

1. Upon the initiation of each runtime BIT, the entry and exit times for the processing.

2. Upon completion of the parameter initialization function, the name of the dataset that defines the parameters.

3. Whenever the operator changes any parameter, or injects a fault, the variable name and value.

4. The state of communications with the external computer whenever it changes.

5. An alert issued to the operator.

6. Whenever an alert is removed from the alert list, the time of alert removal.

7. When the runtime BIT processing is initiated, the time of BIT initiation.

Whenever a value is recorded, it shall be time-tagged with the current system time. If a write error is detected while recording data that is to be extracted, an alert shall be issued to the operator, and the data extract capability shall be set to an error state.

# References

1. DoD Instruction 5000.31, July 1986.

2. B. Craig Meyers and Nelson H. Weiderman, System Specification Document: Shipboard Inertial Navigation System Simulator and External Computer, October 1988, CMU/SEI-88-TR-24.

3. Program Performance Specification for the Inertial Navigation Set AN/WSN-5, May 1985.

4. Program Design Specification for the Inertial Navigation Set AN/WSN-5, May 1985.

5. Interface Design Specification for the Inertial Navigation Set AN/WSN-5 to external computer, NAVSEA T9427-AA-IDS-010/WSN-5, August 1982.

6. Charles F. Fennemore and Terry L. Harter, Alignment Computations for Vertically Launched Tomahawk Missiles, NSWC TR 83-113, July 1983.

7. B. Craig Meyers and Hans Mumm, Performance Specification for an External Computer System Simulator, CMU/SEI-88-TR-25, October 1988.

8. Charles F. Fennemore and Terry L. Harter, Motion Characteristics of Ships Equipped with Vertically Launched Tomahawk Missiles, NSWC TR-83-433, December 1983.

# Appendix 1: List of Acronyms

| | |
|---|---|
| a | lever arm constant |
| ACK | acknowledge |
| $A_{heave}$ | amplitude of ship heave motion |
| $A_{pitch}$ | amplitude of ship pitch motion |
| $A_{roll}$ | amplitude of ship roll motion |
| $A_{surge}$ | amplitude of ship surge motion |
| $A_{sway}$ | amplitude of ship sway motion |
| $A_{yaw}$ | amplitude of ship yaw motion |
| b | lever arm constant |
| BIT | built-in test |
| c | lever arm constant |
| DX | data extraction |
| EC | external computer |
| EF | external function |
| EOM | end of message |
| GMT | Greenwich Mean Time |
| INS | inertial navigation system |
| $M_{heave}$ | ship motion due to heave |
| $M_{pitch}$ | ship motion due to pitch |
| $M_{roll}$ | ship motion due to roll |
| $M_{surge}$ | ship motion due to surge |
| $M_{sway}$ | ship motion due to sway |
| $M_{yaw}$ | ship motion due to yaw |
| NAK | negative acknowledge |
| $R_e$ | radius of the earth |
| $R_{pitch}$ | rate of change of ship pitch motion |
| RTR | ready to receive |
| s | ship speed |
| SOM | start of message |
| SOTM | start of test message |
| TGR | time of gyro reset |
| $V_E$ | east component of ship velocity |
| $V_N$ | north component of ship velocity |
| $V_Z$ | vertical component of ship velocity |
| $\alpha$ | ship heading |
| $\alpha_{rate}$ | rate of change of ship heading |
| $\beta$ | commanded ship course |
| $\eta$ | longitude |
| $\theta_{list}$ | ship list |
| $\theta_{trim}$ | ship trim |
| $\phi$ | latitude |
| $\phi_{heave}$ | phase constant of ship heave motion |
| $\phi_{pitch}$ | phase constant of ship pitch motion |

| | |
|---|---|
| $\phi_{roll}$ | phase constant of ship roll motion |
| $\phi_{surge}$ | phase constant of ship surge motion |
| $\phi_{sway}$ | phase constant of ship sway motion |
| $\phi_{yaw}$ | phase constant of ship yaw motion |
| $\omega_{heave}$ | circular frequency of ship heave motion |
| $\omega_{pitch}$ | circular frequency of ship pitch motion |
| $\omega_{roll}$ | circular frequency of ship roll motion |
| $\omega_{surge}$ | circular frequency of ship surge motion |
| $\omega_{sway}$ | circular frequency of ship sway motion |
| $\omega_{yaw}$ | circular frequency of ship yaw motion |

# Appendix 2: Calculation of Ship Roll and Roll Rate

The calculation of the ship roll motion $M_{roll}$ shall be based on the following equation:

$$M_{roll} = A_{roll} \sin(\omega_{roll} t + \phi_{roll}) + \theta_{list}$$

In the above equation:

1. $A_{roll}$ represents the amplitude of the ship roll motion in degrees.
2. $\omega_{roll}$ represents the circular frequency of the ship roll motion in radians per second.
3. $\phi_{roll}$ represents the phase of the ship roll motion.
4. $\theta_{list}$ represents the value of the ship list.

The ship roll rate $R_{roll}$ is obtained by a differentiation of the above with respect to time, giving:

$$R_{roll} = A_{roll} \omega_{roll} \cos(\omega_{roll} t + \phi_{roll})$$

.

# Appendix 3: Calculation of Ship Pitch and Pitch Rate

The calculation of ship pitch motion $M_{pitch}$ shall be based on the following equation:

$$M_{pitch} = A_{pitch} \sin(\omega_{pitch} t + \phi_{pitch}) + \theta_{trim}$$

In the above equation:

1. $A_{pitch}$ represents the amplitude of the pitch motion in degrees.
2. $\omega_{pitch}$ represents the circular frequency of the pitch motion in radians per second.
3. $\phi_{pitch}$ represents the phase of the pitch motion.
4. $\theta_{trim}$ represents the ship trim.

The ship pitch rate $R_{pitch}$ is obtained by a differentiation of the above with respect to time, giving:

$$R_{pitch} = A_{pitch}\omega_{pitch}\cos(\omega_{pitch}t + \phi_{pitch})$$

# Appendix 4: Calculation of Ship Yaw Motion

The calculation for the ship yaw motion $M_{yaw}$ shall be based on the following sinuosidal approximation:

$$M_{yaw} = A_{yaw} \sin( \omega_{yaw} t + \phi_{yaw} )$$

In the above equation:

1. $A_{yaw}$ denotes the amplitude of the ship yaw motion in degrees.
2. $\omega_{yaw}$ represents the circular frequency of the ship yaw motion in radians per second.
3. $\phi_{yaw}$ represents the phase of the ship yaw motion.

# Appendix 5: Calculation of Ship Surge Motion

The calculation for the ship surge motion $M_{surge}$ shall be based on the following sinuosidal approximation:

$$M_{surge} = A_{surge} \sin ( \omega_{surge} t + \phi_{surge} )$$

In the above equation:

1. $A_{surge}$ represents the amplitude of the surge motion in feet.
2. $\omega_{surge}$ represents the circular frequency of the surge motion in radians per second.
3. $\phi_{surge}$ represents the phase of the surge motion.

# Appendix 6: Calculation of Ship Sway Motion

The calculation for the ship sway motion $M_{sway}$ shall be based on the following sinuosidal approximation:

$$M_{sway} = A_{sway} \sin( \omega_{sway} \, t + \phi_{sway} )$$

In the above equation:

1. $A_{sway}$ represents the amplitude of the sway motion in feet.
2. $\omega_{sway}$ represents the circular frequency of the sway motion in radians per second.
3. $\phi_{sway}$ represents the phase of the sway motion.

# Appendix 7: Calculation of Ship Heave Motion

The calculation for the ship heave $M_{heave}$ motion shall be based on the following sinuosidal approximation:

$$M_{heave} = A_{heave} \sin( \omega_{heave} t + \phi_{heave} )$$

In the above equation:

1. $A_{heave}$ represents the amplitude of the ship heave motion in feet.
2. $\omega_{heave}$ represents the circular frequency of the ship heave motion in radians per second.
3. $\phi_{heave}$ represents the phase of the ship heave motion.

# Appendix 8: Calculation of Ship Heading and Heading Rate

The ship heading and heading rate are required to be updated on a periodic basis. Let $\beta$ denote the commanded ship course. Then, the ship heading $\alpha$ is defined by:

$$\alpha = \beta + M_{yaw}$$

where $M_{yaw}$ is the ship yaw motion. (See Appendix 4.)

The heading rate to be denoted $\alpha_{rate}$ shall be calculated from the following:

$$\alpha_{rate} = \frac{d}{dt}(\beta + M_{yaw})$$

In the above, the rate of change of the commanded course is either zero or the turn rate defined by the operator; see Section 3.3.2.5. The rate of change of the yaw motion may be obtained from Appendix 4.

# Appendix 9: Ship Position Update Calculations

The INS shall be able to calculate updates to the position, specified by latitude $\phi$ and longitude $\theta$. The change in position is a function of the ship heading $\alpha$ and the ship speed s, according to the following two cases:

**Case 1:** Absolute value of heading $\alpha$ not equal to $\pi/2$:

$$\phi_2 = \phi_1 + \left(\frac{st}{R_e}\right) \cos \alpha$$

$$\theta_2 = \theta_1 + (\tan \alpha) \, Ln\left\{\frac{\tan[(\phi_2/2)+\pi/4]}{\tan[(\phi_1/2)+\pi/4]}\right\}$$

**Case 2:** Absolute value of heading $\alpha = \pi/2$:

$$\phi_2 = \phi_1$$

$$\theta_2 = \theta_1 + \frac{st}{R_e \cos \phi_1}$$

In the above:

$(\phi_1, \theta_1)$ = initial position

$(\phi_2, \theta_2)$ = final position

$R_e$ = radius of the earth

$t$ = time interval between updates

$s$ = ship speed

$\alpha$ = ship heading

# Appendix 10: Calculation of Ship Velocity Components

This appendix describes the equations to be used in the calculation of the ship velocity components. Additionally, the calculation of the integrated east and north components of velocity are also presented here. The calculations described below are required for the ship velocity update function, described in Section 3.11.

Calculation of the translational velocities are based on the following equation:

$$
\begin{bmatrix} V_N \\ V_E \\ -V_Z \end{bmatrix} = L_3(-\beta) \begin{bmatrix} s \\ 0 \\ 0 \end{bmatrix} + L_3(-\alpha) \begin{bmatrix} V_{surge} \\ V_{sway} \\ V_{heave} \end{bmatrix}
$$

$$
+ L_3(-\alpha) L_2(\gamma_p) L_1(\gamma_r)
$$

$$
\frac{\pi}{180} \left\{ \left( \begin{bmatrix} 1 & 0 & \sin \gamma_p \\ 0 & \cos \gamma_r & -\cos \gamma_p \sin \gamma_r \\ 0 & \sin \gamma_r & \cos \gamma_p \cos \gamma_r \end{bmatrix} \begin{bmatrix} -R_{roll} \\ -R_{pitch} \\ \alpha_{rate} \end{bmatrix} \right) \otimes \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\}
$$

where,

$\beta$ = commanded course,

$\alpha$ = ownship heading,

$s$ = ownship speed,

$V_E$ = East velocity component,

$V_N$ = North velocity component,

$V_Z$ = Vertical velocity component,

$V_{surge}$ = Surge velocity,

$V_{sway}$ = Sway velocity,

$V_{heave}$ = Heave velocity,

$\gamma_p$ = ownship pitch,

$\gamma_r$ = ownship roll,

$R_{roll}$ = ownship roll rate,

$R_{pitch}$ = ownship pitch rate,

$\alpha_{rate}$ = ownship heading rate,

$a, b, c$ = lever arm constants,

$\otimes$ = vector cross product.

The rotation matrices $L_i(\theta)$ are defined by:

$$L_1(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}$$

$$L_2(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$L_3(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A derivation of the above result is not included here. However, a similar computation is documented in detail in Section 6.0 of reference [6].

The integrated North and East velocities may be obtained by numerical integration. Thus, let $f(t_i)$ denote either of these quantities at time $t_i$. Then,

$$f(t_{i+1}) = f(t_i) + v_i \, \Delta t,$$

where $v_i$ is the respective velocity component and $\Delta t$ denotes the time interval $[t_i, t_{i+1}]$.

# Appendix 11: Lever Arm Constants

To compute translational velocities, discussed in Appendix 10, it is necessary to consider the fact that the WSN-5 is not located at the ship center of gravity. Thus, one defines a lever arm vector having components (a, b, c) that are defined by:

1. a = alongship distance (in feet) from the ship center of gravity to the WSN-5. The positive direction is toward forward, and negative is toward aft.
2. b = athwartship distance (in feet) from the ship center of gravity to the WSN-5. The positive direction is toward starboard, and negative is toward the port side.
3. c = vertical distance (in feet) from the ship center of gravity to the WSN-5. The positive direction is down and the negative direction is up.

The lever arm constants defined above are referred to as ship "adaptation data." That is, the values of these parameters will vary from ship class to ship class. For purposes of this document, the valid ranges of these parameters are defined as follows:

1. The value of the parameter "a" must be in the range -250 feet to 250 feet.
2. The value of the parameter "b" must be in the range -25 feet to 25 feet.
3. The value of the parameter "c" must be in the range -25 feet to 25 feet.

The values of the lever arm constants are obtained by the Parameter Initialization Function; see Section 3.2.

# Appendix 12: Default Values of Parameters as a Function of Sea State

The operator shall have the ability to specify default values of the basic simulation parameters. Additionally, these parameters shall be characterized by sea state and shall include values for sea state 1 through sea state 7. Table 3 lists the values of the default parameters as a function of sea state. The parameters listed are typical for a CG-47 class ship in a long-crested sea; see reference [8].

**Table 3:** Default Values of Parameters as a Function of Sea State

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $A_{heave}$ | 0.0 | 0.5 | 0.8 | 1.2 | 2.1 | 4.9 | 10.7 |
| $\omega_{heave}$ | 0.0 | 0.21 | 0.27 | 0.32 | 0.37 | 0.37 | 0.37 |
| $\phi_{heave}$ | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| $A_{pitch}$ | 0.0 | 0.2 | 0.4 | 0.7 | 1.0 | 1.7 | 2.9 |
| $\omega_{pitch}$ | 0.0 | 0.21 | 0.27 | 0.32 | 0.37 | 0.37 | 0.37 |
| $\phi_{pitch}$ | 60.0 | 60.0 | 60.0 | 60.0 | 60.0 | 60.0 | 60.0 |
| $A_{roll}$ | 0.0 | 1.0 | 3.0 | 5.0 | 13.4 | 19.5 | 28.6 |
| $\omega_{roll}$ | 0.0 | 0.21 | 0.27 | 0.32 | 0.37 | 0.37 | 0.37 |
| $\phi_{roll}$ | 120.0 | 120.0 | 120.0 | 120.0 | 120.0 | 120.0 | 120.0 |
| $A_{sway}$ | 0.0 | 1.0 | 1.8 | 2.6 | 4.5 | 7.9 | 14.8 |
| $\omega_{sway}$ | 0.0 | 0.21 | 0.27 | 0.32 | 0.37 | 0.37 | 0.37 |
| $\phi_{sway}$ | 180.0 | 180.0 | 180.0 | 180.0 | 180.0 | 180.0 | 180.0 |
| $A_{yaw}$ | 0.0 | 0.3 | 0.6 | 0.9 | 2.1 | 3.2 | 5.0 |
| $\omega_{yaw}$ | 0.0 | 0.21 | 0.27 | 0.32 | 0.37 | 0.37 | 0.37 |
| $\phi_{yaw}$ | 240.0 | 240.0 | 240.0 | 240.0 | 240.0 | 240.0 | 240.0 |
| $A_{surge}$ | 0.0 | 1.0 | 1.9 | 3.0 | 3.2 | 5.4 | 9.7 |
| $\omega_{surge}$ | 0.0 | 0.21 | 0.27 | 0.32 | 0.37 | 0.37 | 0.37 |
| $\phi_{surge}$ | 300.0 | 300.0 | 300.0 | 300.0 | 300.0 | 300.0 | 300.0 |

## REPORT DOCUMENTATION PAGE

| ..REPORT SECURITY CLASSIFICATION<br>UNCLASSIFIED | | 1b. RESTRICTIVE MARKINGS<br>NONE | | | |
|---|---|---|---|---|---|
| .. SECURITY CLASSIFICATION AUTHORITY<br>N/A | | 3. DISTRIBUTION/AVAILABILITY OF REPORT<br>APPROVED FOR PUBLIC RELEASE<br>DISTRIBUTION UNLIMITED | | | |
| ). DECLASSIFICATION/DOWNGRADING SCHEDULE<br>N/A | | | | | |
| PERFORMING ORGANIZATION REPORT NUMBER(S)<br>CMU/SEI-88-TR-23 | | 5. MONITORING ORGANIZATION REPORT NUMBER(S)<br>ESD-TR-88-024 | | | |
| .. NAME OF PERFORMING ORGANIZATION<br>SOFTWARE ENGINEERING INSTITUTE | 6b. OFFICE SYMBOL<br>(If applicable)<br>SEI | 7a. NAME OF MONITORING ORGANIZATION<br>SEI JOINT PROGRAM OFFICE | | | |
| .. ADDRESS (City, State and ZIP Code)<br>CARNEGIE MELLON UNIVERSITY<br>PITTSBURGH, PA 15213 | | 7b. ADDRESS (City, State and ZIP Code)<br>ESD/XRS1<br>HANSCOM AIR FORCE BASE, MA 01731 | | | |
| .. NAME OF FUNDING/SPONSORING<br>ORGANIZATION<br>SEI JOINT PROGRAM OFFICE | 8b. OFFICE SYMBOL<br>(If applicable)<br>SEI JPO | 9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER<br>F1962885C0003 | | | |
| .. ADDRESS (City, State and ZIP Code)<br>CARNEGIE MELLON UNIVERSITY<br>SOFTWARE ENGINEERING INSTITUTE JPO<br>PITTSBURGH, PA 15213 | | 10. SOURCE OF FUNDING NOS. | | | |
| | | PROGRAM<br>ELEMENT NO. | PROJECT<br>NO.<br>N/A | TASK<br>NO.<br>N/A | WORK UNIT<br>NO.<br>N/A |

1. TITLE (Include Security Classification)
Functional Performance Specification for an Inertial Navigation System

2. PERSONAL AUTHOR(S)
B. Craig Meyers and Nelson H. Weiderman

| 13a. TYPE OF REPORT<br>FINAL | 13b. TIME COVERED<br>FROM _____ TO _____ | 14. DATE OF REPORT (Yr., Mo., Day)<br>August 1988 | 15. PAGE COUNT<br>59 |
|---|---|---|---|

16. SUPPLEMENTARY NOTATION

| 17. COSATI CODES | | | 18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) |
|---|---|---|---|
| FIELD | GROUP | SUB. GR. | functional performance specification<br>inertial navigation system<br>external computer system<br>INS |
| | | | |
| | | | |

19. ABSTRACT (Continue on reverse if necessary and identify by block number)
This document defines the functional and performance requirements for the inertial navigation system simulator that interfaces with the external computer system (ECS) simulator. Both the INS simulator and the ECS simulator are being developed in Ada by the Real-Time Embedded Systems Testbed Project at the Software Engineering Institute (SEI) The INS simulator is similar to a real-world INS, but has reduced functionality. This document provides specifications for the major functions on the INS simulator.

| 20. DISTRIBUTION/AVAILABILITY OF ABSTRACT | 21. ABSTRACT SECURITY CLASSIFICATION | |
|---|---|---|
| UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☒ | UNCLASSIFIED, UNLIMITED | |
| 22a. NAME OF RESPONSIBLE INDIVIDUAL<br>KARL SHINGLER | 22b. TELEPHONE NUMBER<br>(Include Area Code)<br>(412) 268-7630 | 22c. OFFICE SYMBOL<br>SEI JPO |